IN THE CLAIMS:

Please amend Claims 1, 9, 27, 31, and 32 as shown in this complete set of all pending Claims:

1. (Currently amended) A clock recovery device comprising:

a data rate input;

an input data sampler that obtains samples of a serial data stream;

an early/late voting logic that identifies early and late operation for a set of bit times of the serial data stream from the obtained samples;

a finite state machine operable to receive the data rate input for bandwidth control of the finite state machine and the early and late operation indication, and select two clock phases from identified operations of the voting logic and generates interpolator control signals to selectively mix these two clock phases;

a phase interpolator operable to select a clock phase output that is generated from the two selected clock phases received from a clock input , wherein the phase interpolator selects the clock phase output based on the interpolator control signals from the finite state machine; and

a programmable divider that divides the clock phase output of the phase interpolator based on the received data rate to program the device to operate at one of a plurality of clock and data rates, bandwidths, or tracking capabilities.

2. (Original) The device of claim 1, further comprising a rate/mode control circuit operable to receive the data rate input and program a divide ratio of the programmable divider and to control the bandwidth of the finite state machine.

3 and 4. (Canceled)

5. (Original) The device of claim 1, further comprising a frequency detection control circuit operable to automatically detect the clock or data rate and program a divide ratio of the programmable divider and to control the bandwidth of the finite state machine.

6-8. (Canceled)

9. (Currently amended) The device of claim 1, wherein the bandwidth of the clock recovery [[loop]] device can be controlled by selecting multiple phase updates from the phase interpolator [[or by controlling a latency of the updates]].

10. (Original) The device of claim 1, wherein the finite state machine comprises a decode circuit and a plurality of delay elements operable to selectively determine the quantity of phase correction steps per update based on a function of the data rate input, and to connect the delay elements in a configuration to provide the required steps.

11-19. (Canceled)

20. (Original) The device of claim 1, wherein the programmable divider comprises a plurality of dividers connected to a multiplexor, the dividers each having a fixed divide ratio, the multiplexor configured to select an output of one of the dividers based on a function of the data rate input to provide a programmable data rate and a substantially fixed device bandwidth.

21. (Original) The device of claim 20, wherein a portion of the plurality of fixed dividers are serially connected to each other and to the multiplexor, the multiplexor configured to select the output of one of the dividers based on a function of the data rate input to provide a programmable data rate and a substantially fixed device bandwidth.

22. (Original) The device of claim 1, wherein the programmable divider functionally resides after the interpolator.

23. (Original) The device of claim 1, wherein the programmable divider functionally resides between the interpolator and the input data sampler.

24-26. (Canceled)

27. (Currently amended) A method of clock recovery comprising:

receiving a data rate input;

obtaining samples of a received serial data stream for a set of consecutive bit times according to a data clock and a transition clock;

setting a programmable divider rate and finite state machine bandwidth based on the received data rate;

analyzing the set of consecutive bit times to identify early, late, and neither conditions of the clock within the respective consecutive bit times according to the obtained samples;

inputting a plurality of clock phases;

determining two of the plurality of clock phases based on the identified early, late or neither clock condition;

adjusting a clock phase correction step size based on the data rate;

interpolating between the two [[indicated]] determined clock phases based on [[the]] a relative time difference between [[the]] a prior and current phase transition indication; and

correcting [[the]] a clock phase error according to the adjusted step size.

28. (Original) The method of claim 27, further comprising:

latching the data and synchronizing the clock with the corrected clock phase.

29. (Original) The method of claim 28, further comprising:

outputting a recovered clock signal and a recovered data signal.

30. (Original) The method of claim 27, wherein inputting a plurality of clock phases comprises inputting a plurality of clock phases with fixed frequency.

31. (Currently amended) The method of claim 27, wherein obtaining [[center]] samples comprises sampling the received serial data stream on rising transitions of a data clock.

32. (Currently amended) The method of claim 27, wherein obtaining [[transition]] samples comprises sampling the received serial data stream on rising transitions of a transition clock.

33. (Original) The method of claim 27, wherein analyzing the set of consecutive bit times to identify late operation(s) comprises identifying a transition between a current transition sample and a previous center sample for each bit time.

34. (Original) The method of claim 27, wherein analyzing the set of consecutive bit times to identify early operation(s) comprises identifying a transition between a current transition sample and a current center sample for each bit time.

35. (Original) A method of preventing interpolator update glitching comprising:

receiving a data rate input, and a voting operation indication for a required phase correction;

decoding and selecting a plurality of partial phase corrections for the phase correction based on the received data rate;

enabling a plurality of delay elements corresponding to the plurality of partial phase corrections in response to the required phase correction; and

generating a plurality of successive partial phase corrections during an interpolator update to provide the required phase correction.

36. (Original) The method of claim 35, wherein the decoding and selecting a plurality of partial phase corrections comprises decoding a 32 bit thermometer code of delay elements to provide 32 partial phase correction steps in a sequential string.